

MANAGEMENT OF COMPUTER SERVERS

FIELD OF THE INVENTION

[0001]One embodiment of the present invention is directed to computer servers. More particularly, one embodiment of the present invention is directed to the management of computer servers.

BACKGROUND INFORMATION

[0002]Computer networks are exploding in use, due in part to the popularity of the Internet. The content of the Internet is typically stored on computer servers ("Web servers") that store information, and serve the information to other computers on the Internet in response to a remote request. Computer servers on the Internet and other computer networks also are used for a myriad of other tasks that form the basic and advanced plumbing of any network. Examples of these tasks include Dynamic Host

configuration ("DHCP") for dynamic Internet Protocol ("IP") address assignment, Dynamic Name Service ("DNS") for domain name resolution to an IP address (e.g., "yahoo.com" to "203.3.42.2"), and E-mail functionality.

[0003] Each server in a network from time to time must be maintained or "managed". For example, a new record might have to be added to a DNS server, or a record may have to be modified. Many of the servers that form the Internet operate under the Linux or UNIX operating systems. Services on a Linux or UNIX server are typically maintained by modifying configuration files that reside in the "/etc" directory of the server. Currently, when a change needs to be made to a particular service, such as DHCP or E-mail, one of the associated files, usually denoted with the extension of .conf, must be modified manually, on a per-server basis, with a text editor and then saved. Once the changes have been saved, the associated service must be restarted to cause the changes to take effect.

[0004] The known method of managing multiple configuration files for each service on every server requires a large amount of time and skill level. Care must be taken to ensure that the configuration is done properly or else the operator must go through a slow process of parsing the changes to find the flaw.

[0005] Based on the foregoing, there is a need for a improved system and method for managing computer servers that reduces the complexity and time required by known methods and systems.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] Fig. 1 illustrates an example of a computer network that can implement one embodiment of the present invention.

[0007] Fig. 2 is a graphical view of an interface executed by a client computer of Fig. 1 in accordance with one embodiment of the present invention.

[0008] Fig. 3 illustrates an overview of the interaction between an interface of a client computer, an agent manager of a management server and each server agent of managed servers.

DETAILED DESCRIPTION

[0009] One embodiment of the present invention is a computer server management system that allows multiple servers to be managed at the same time from a client computer having a graphical user interface ("GUI"). The result is an easy-to-use and intuitive interface for the management of multiple network services across multiple servers.

[0010] Fig. 1 illustrates an example of a computer network 10 that can implement one embodiment of the present invention. Network 10 includes a client computer 12, a management server computer 14, and managed server computers 20-23. All computers in network 10 are coupled together through communication links. The communication links may be local links, or may be remote links, such as communication links through the Internet.

[0011]One embodiment of client computer 12 executes a client application GUI or any other type of user interface 13 that allows managed tasks to be initiated. Client computer 12 can be any general purpose computer capable of executing software for implementing a GUI. The client application GUI is an intuitive, cross-platform application that allows secure remote access to the servers and services under management. Multiple client computers can allow multiple users to manage the remote servers on network 10.

[0012]Managed servers 20-23 are general purpose computer servers that in one embodiment execute Internet Web services. In one embodiment, one or more of managed servers 20-23 execute the Linux or UNIX operating system. Embodiments of managed servers 20-23 also each execute a server agent. In one embodiment, the server agent is a software program that resides on each managed servers 20-23 as a server daemon with its own user account with administrator privileges in order to manage each of the network services on the particular server.

[0013]One embodiment of management server 14 is a general purpose computer server that executes an agent manager 15. Agent manager 15 is responsible for communication between interface 13 on client computer 12 and the server agents on managed servers 20-23. Management server 14 is coupled to a management database 16. One embodiment of management database 16 stores all current configurations of managed servers 20-23 and a select amount of previous configurations for reference purposes.

[0014] Agent manager 15 of management server 14 provides the method of communication between the individual server agents on managed servers 20-23 under management and interface 13 on client computer 12, while maintaining management database 16 with the most current information and archiving any older information for rollback needs.

[0015] Management database 16 is the central location where all necessary configuration information is stored and referenced by both managed servers 20-23 and the client interfaces. The communication between each client interface is controlled through agent manager 15. Management database 16 stores *.conf files of each service on a per server basis and maintains backup copies of old configurations for roll back purposes. Management database 16 further maintains per server configuration information for both the client interface and the managed server agent.

[0016] One embodiment of management database 16 maintains several tables that allows management server 14 to keep track of multiple variables necessary to maintain a network of servers and services securely. Some of these tables may include a user table to provide for authentication separate from the server user/password files so that permissions to services can be passed in the authentication messages more easily. This separation of authentication into the database also simplifies the management of user accounts while providing for full resiliency if a network or system failure occurs. In one embodiment, this resiliency is attainable as long as the database is frequently backed up or replicated.

[0017] Each embodiment of the agent on each managed server 20-23 is responsible for the local management of the individual server on which it resides. It makes all necessary changes to the individual configuration files that reside in the directory for each network service application. The configuration files are typically located in the /etc directory or some subdirectory within the /etc directory in one embodiment. In other embodiments, the configuration files are stored in another directory.

[0018] In another embodiment of network 10, the functionality of agent manager 15 and database 16 is implemented on client computer 12, and management server 14 is eliminated. This "client-only" embodiment can be used when a small number (e.g., 10 or less) of servers are being managed. In contrast, the "client/server" model illustrated in Fig. 1 can be used to manage a larger number (e.g., 50-75) of servers and services at once.

[0019] Fig. 2 is a graphical view of an interface 48 executed by client computer 12 of Fig. 1 in accordance with one embodiment of the present invention. In one embodiment, interface 48 resides on the respective client computer of each individual user that is managing managed servers 20-23. In one embodiment, interface 48 is a GUI and is split into different panes or bars 40 and 45 in a column view. Each bar is responsible for providing an easy-to-use application by separating the larger management tasks into smaller distinct categories based on server application, server name, or server group. For each client computer running interface 48, an Extensible

Markup Language (“XML”) file may be stored locally containing the last known list of servers and services that are being managed.

[0020] Interface 48 includes a management service bar 40 where the individual services that need to be managed are chosen. Each service that can be managed may be displayed as a separate icon. Access to these services are allowed during the authentication process during login. This is to allow the capability of assigning the management responsibility of certain services to different users. For example, different groups in an organization may be responsible for the management of Web services such as HyperText Transfer Protocol (“HTTP”) and DNS, while others could be responsible for the management of end-user services such as Dynamic Host Configuration Protocol (“DHCP”), File Transfer Protocol (“FTP”), Samba, Lightweight Directory Access Protocol (“LDAP”), and E-mail.

[0021] Once a service is chosen from the first column, or management service bar 40, a list of servers currently under management on the client application will appear in a server list bar 45. This allows the end-user to see only the servers they are responsible for in server list bar 45.

[0022] Server list bar 45 provides a list of servers that are configured to run the service selected from management service bar 40. Servers can be grouped together in several ways. For example, servers can be grouped by organization (e.g., Finance), or by geographic region (e.g., Southeast U.S.).

[0023] Servers may need to be added for each service. To ease the addition and deletion of servers on a per service basis, servers that need to be managed can be added to interface 48 globally under a managed servers component that may appear in the management service bar 40. By adding the individual servers globally, the servers can be added individually under each service more easily. During the addition of any new server, multiple options may be provisioned. Examples of options include Server Type, Host name, IP Address, DNS name, system hardware configuration auto discovery, notes regarding an individual server, etc.

[0024] Interface 48 further includes a menu 50 that allows information to be entered in order to execute the desired management task. Menu 50 will vary depending on which service is selected and which server or servers are selected. Menu 50 includes tabs for viewing additional information.

[0025] Fig. 3 illustrates an overview of the interaction between interface 13 of client computer 12, agent manager 15 of management server 14 and each server agent of managed servers 20-23. The interaction shown in Fig. 3 provides an example of a management change from login to first change, and includes interactions 100, 200 and 300.

[0026] Interactions 100: Upon instantiation of the server agent on the managed server, either due a new install or system reboot, the agent will make a service request

of agent manager 15 to verify what services are configured for the server it is managing.

In addition, the server agent will check if there are any recent changes

[0027]Interactions 200: In order to verify the user's credentials and permissions, each user is required to login to agent manager 15 in one embodiment. This provides the additional benefit of ensuring that the user is authenticated to manage a given service. In one embodiment, each user will be set up by the administrator account with permissions to the services they are responsible for before they may login and use the interface 13.

[0028]Interactions 300: When a service needs to be added or modified, interface 13 will send a request ("Most recent config") for the current *.conf file for that service on the respective server which is then requested from agent manager 15. Upon reception of the file ("Config file") any changes to the service are applied to the *.conf file ("Modify Request") and sent back to agent manager 15 for replication to the server agents ("Service update"). In one embodiment, agent manager 15 communicates the changes to the server agents using simple messaging, informing the server agents of any modifications needed for a particular service application. A success or failure notice is then sent by the server agents to agent manager 15, and then to interface 13.

[0029]As described, the server management system in accordance with embodiments of the present invention allows network administrators and other users to manage multiple servers and multiple services across a network or within a single data

center. The matrix design of the client interface gives the user the ability to manage multiple services that exist on multiple servers at one time.

[0030] Embodiments of the invention can be deployed to address a broader market due flexible scaling and cost structure. In addition, managed server agent software can be installed onto network servers with little or no service disruption. Embodiments are implemented so that the agent software operates as a user on the managed systems, allowing for security through a self contained environment on each server being managed.

[0031] Several embodiments of the present invention are specifically illustrated and/or described herein. However, it will be appreciated that modifications and variations of the present invention are covered by the above teachings and within the purview of the appended claims without departing from the spirit and intended scope of the invention.